An ALE ESFEM for solving PDEs on evolving surfaces

Article (Submitted Version)

An ALE ESFEM for solving PDEs on evolving surfaces

Charles M. Elliott and Vanessa Styles

Abstract. Numerical methods for approximating the solution of partial differential equations on evolving hypersurfaces using surface finite elements on evolving triangulated surfaces are presented. In the ALE ESFEM the vertices of the triangles evolve with a velocity which is normal to the hypersurface whilst having a tangential velocity which is arbitrary. This is in contrast to the original evolving surface finite element method in which the nodes move with a material velocity. Numerical experiments are presented which illustrate the value of choosing the arbitrary tangential velocity to improve mesh quality. Simulations of two applications arising in material science and biology are presented which couple the evolution of the surface to the solution of the surface partial differential equation.

Mathematics Subject Classification (2010). Primary 65M60, 65M15; Secondary 35K99, 35R01, 35R37, 76R99.

Keywords. Surface finite elements, ALE, advection diffusion equation, interface motion.

1. Introduction

We propose an ALE ESFEM (Arbitrary Eulerian Lagrangian evolving surface finite element method) for the numerical solution of parabolic equations on evolving hypersurfaces $\Gamma(t)$. In particular we consider advection-diffusion equations of the form

$$\partial^\bullet u \ - \mathcal{D}\Delta_\Gamma u + u\nabla_\Gamma \cdot \mathbf{v} = f \text{ on } \Gamma(t). \tag{1.1}$$

The work of V. Styles was supported by the UK Engineering and Physical Sciences Research Council EPSRC Grant EP/D078334/1. The work of C. M. Elliott was supported by the UK Engineering and Physical Sciences Research Council EPSRC Grant EP/G010404.

Here $\Gamma(t)$ is an evolving two-dimensional hypersurface in $\mathbb{R}^3$ with unit normal $\boldsymbol{\nu}$ and normal velocity $\mathbf{v}_\nu = V\boldsymbol{\nu}$. Furthermore

$$\partial^\bullet u \ := \frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u$$

denotes the material derivative of u, $\mathbf{v} = \mathbf{v}_\nu + \mathbf{v}_\tau$, $\mathbf{v}_\tau$ is an advective tangential velocity field, $\mathcal{D} > 0$ is a constant scalar diffusivity and $\Delta_\Gamma u := \nabla_\Gamma \cdot \nabla_\Gamma u$ is the Laplace Beltrami operator (or surface Laplacian), with

$$\nabla_\Gamma u := \nabla u - \nabla u \cdot \boldsymbol{\nu}\boldsymbol{\nu}$$

denoting the tangential gradient. We call $\mathbf{v}$ the material velocity. We rewrite the material derivative as

$$\partial^\bullet u = \partial^\circ u + \mathbf{v}_\tau \cdot \nabla_\Gamma u$$

where we set

$$\partial^\circ u = u_t + \mathbf{v}_\nu \cdot \nabla u$$

to be the normal time derivative.

1.1. Moving hypersurface

By an evolving hypersurface we mean that $\Gamma(t)$ is defined to be the zero level set of a level set function $\varphi : \mathcal{U} \times (0,T) \to \mathbb{R}$ where $\mathcal{U}$ is an open set in $\mathbb{R}^3$ so

$$\Gamma(t) := \{x \in \mathcal{U} : \varphi(x,t) = 0\}$$

and $\nabla\varphi(x,t) \neq 0,\ x \in \Gamma(t)$. For such a hypersurface we define an oriented normal by

$$\nu(x,t) = \frac{\nabla\varphi(x,t}{|\nabla\varphi(x,t)|}$$

and the normal velocity $V\nu$ by

$$V(x,t) := -\frac{\varphi_t(x,t)}{|\nabla\varphi(x,t)|}.$$

Thus moving a point $P_0 \in \Gamma(0)$ by the velocity

$$\dot{P} = V(P,t)\nu(P,t) + \mathbf{v}_\tau^a(P,t),\ P(0) = P_0$$

where $\mathbf{v}_\tau^a$ is an arbitrary tangential velocity field satisfying

$$\mathbf{v}_\tau^a(x,t) \cdot \nu(x,t) = 0$$

keeps $P(t)$ on the surface $\Gamma(t)$ because

$$\frac{d}{dt}\varphi(P(t),t) = \nabla\varphi(P,t) \cdot \dot{P} + \varphi_t(P,t) = 0.$$

Thus the hypersurface $\Gamma(t)$ may also be described by a map $\mathbf{X} : \Gamma(0) \times (0,T) \to \mathbb{R}^3$ so that

$$\Gamma(t) = \{x = \mathbf{X}(P_0,t), P_0 \in \Gamma(0)\}$$

and

$$\mathbf{X}_t \cdot \nu = V.$$

This map may be used to evolve the nodes of the triangulation.The tangential velocity, $\mathbf{X}_t - V\nu$, of the map may be chosen in several ways:

- There may be a material velocity $\mathbf{v} = \mathbf{v}_\nu + \mathbf{v}_\tau$ in the mathematical model with which we wish to define the map. For example this might arise when the hypersurface is a fluid material interface and choosing $\mathbf{X}_t = \mathbf{v}$ implies that points $\mathbf{X}(P_0, t), P_0 \in \Gamma_0$, evolve as material points.
- It may be that there is natural partial differential equation which evolves a parametrisation such that the hypersurface satisfies a geometric evolution law. For example, the solution of the equation

$$\mathbf{X}_t = \Delta_\Gamma \mathbf{X}$$

defines motion by mean curvature and for which the solution moves in a normal direction. On the other hand the equation

$$\mathbf{X}_t = \frac{\mathbf{X}_{\theta\theta}}{|\mathbf{X}_\theta|^2},\ \mathbf{X}(\theta + 2\pi, t) = \mathbf{X}(\theta, t),\ \theta \in [0, 2\pi)$$

evolves a closed planar curve in the normal direction with velocity given by the curvature but has a tangential velocity defined by the equation.
- It may be that we wish for mathematical or numerical reasons to choose a tangential velocity in such a way as to yield a *nice* map. This is relevant in the context of using triangulated surfaces and evolving triangle vertices so that the triangles do not degenerate.

1.2. ESFEM

The evolving surface finite element method introduced [**?**] uses finite elements on a triangulated surface interpolating $\Gamma(t)$ whose vertices move with the material velocity $\mathbf{v}$. The finite element method is based on the variational form

$$\frac{d}{dt}\int_{\Gamma(t)} u\varphi + \int_{\Gamma(t)} \mathcal{D}\nabla_\Gamma u \cdot \nabla_\Gamma \varphi = \int_{\Gamma(t)} u\partial^\bullet \varphi \tag{1.2}$$

where φ is an arbitrary test function defined on the surface $\Gamma(t)$ for all t. This ESFEM may be viewed as a Lagrangian method. Note that a purely Eulerian approach may be formulated using an implicit surface approach in which an equation holds on all level sets of a prescribed level set function, [**?**, **?**, **?**, **?**].

In this work we propose a variant, ALE ESFEM, of the evolving surface finite element method in which (1.1) is approximated on triangulated surfaces $\Gamma_h(t)$ whose nodes have a velocity which may not be the same as the material velocity. That is the velocity of the nodes are of the form

$$\mathbf{v}_h := V\boldsymbol{\nu} + \mathbf{v}_\tau^a \tag{1.3}$$

where $\mathbf{v}_\tau^a$ is an *arbitrary* tangential velocity, that is $\mathbf{v}_\tau^a \cdot \boldsymbol{\nu} = 0$. We suppose that the hypersurface $\Gamma(t)$ is defined as the zero level set of a prescribed level set function $\varphi : \mathbb{R}^3 \times (0, \infty) \to \mathbb{R}$ then

$$\nu(x,t) = \frac{\nabla\varphi(x,t)}{|\nabla\varphi(x,t)|},\ V(x,t) = \frac{-\varphi_t(x,t)}{|\nabla\varphi(x,t)|}$$

define the normal to $\Gamma(t)$ in the direction of increasing φ and the normal velocity.

Thus it follows from the definition of $\mathbf{v}_h$ that a triangulated surface $\Gamma_h(0)$ interpolating $\Gamma(0)$ is evolved by moving the nodes with the velocity $\mathbf{v}_h$ will yield a moving triangulated surface $\Gamma_h(t)$ which interpolates $\Gamma(t)$. Furthermore, for small time at least, the mesh topology of the triangulation is maintained during the motion.

In this paper we describe this approach in detail, show how different choices of the velocity of the nodes of the triangulated surface $\Gamma_h(t)$ influence the quality of the finite element mesh and how this in turn influences the accuracy of the approximate solutions of (1.1) and apply the method to situations in which the surface itself is computed using a surface finite element method.

1.3. Background: applications

Equations like (1.1) arise in models for many applications in the applied and natural sciences, for example in dealloying metals by surface dissolution, [**?**], diffusion induced grain boundary motion [**?**, **?**, **?**, **?**, **?**], transport of an insoluble surfactant on the interface between two fluids, [**?**, **?**, **?**, **?**], pattern formation on biological surfaces [**?**] and cell migration and chemotaxis, [**?**, **?**].

1.4. Background: numerical methods for surface PDEs

We refer to [**?**, **?**, **?**] for accounts of numerical methods for evolving interfaces and surfaces and in particular to [**?**, **?**, **?**, **?**, **?**] for surface finite element methods for evolving surfaces by geometric flow equations. Let us review briefly the computation of surface partial differential equations and, in particular, parabolic equations on evolving surfaces.

- An important concept is the use of triangulated surfaces on which finite element spaces are constructed and then used in variational formulations of surface PDEs using surface gradients. Such surface piece-wise linear finite elements on triangulated hypersurfaces were proposed and analysed for the Laplace-Beltrami equation by [**?**] and extended to parabolic (including nonlinear and higher order) equations on stationary surfaces in [**?**]. Higher order finite element spaces for elliptic equations were analysed in [**?**], an adaptive finite element method for stationary surfaces was considered in [**?**], coupling surface and bulk elliptic equations was analysed in [**?**] and a discontinuous Galerkin surface finite element method was considered in [**?**].
- The evolving surface finite element method (ESFEM) was introduced by [**?**] in order to treat conservation laws on moving surfaces. The key idea is to use the Leibniz (or transport) formula for the time derivative of integrals over moving surfaces in order to derive weak and variational formulations. The interesting upshot is that the velocity and mean curvature of the surface do not appear explicitly in variational formulations. Further numerical analysis may be found in [**?**, **?**, **?**]. Applications to complex physical and biological models may be found in [**?**, **?**, **?**, **?**].

- An extension of the idea of the surface finite element method (SFEM) is to use surface finite volumes. An analysis of elliptic equations using general meshes is given in [**?**, **?**]. A method for parabolic equations on stationary surfaces using logically Cartesian grids is presented in [**?**]. See also [**?**] for conservation laws on the sphere. Finite volumes may also be used on evolving surfaces, [**?**, **?**].
- Another approach is to numerically solve bulk equations in one space dimension higher. This may be a natural approach when the surface is computed implicitly using phase field or level set methods or when one wishes to use bulk finite element codes. There are a number of variants:
  - One idea is to solve the surface partial differential equations on all level sets of a prescribed function yielding degenerate bulk PDEs. See [**?**, **?**, **?**, **?**] for stationary surfaces. In the context of conservation laws on moving surfaces this is inherently an Eulerian approach see [**?**, **?**] where level set approximations to surface quantities such as the mean curvature and normal velocity were required. On the other hand an elegant formulation avoiding the need to do this was provided in [**?**] using an implicit surface version of the Leibniz formula. The idea is then to exploit the implicit formulation and rather than a surface triangulation use a bulk triangulation which is independent of the surface. In practice it may be useful to solve in a narrow band and use unfitted bulk finite elements, [**?**]. For surface elliptic equations, [**?**] gave a discretisation error analysis for a narrow band level set method using the unfitted finite element method.
  - Another method using a bulk unfitted mesh and a finite element space independent of the surface has been proposed in [**?**, **?**]. Here the variational equation is formulated using tangential gradients on the zero level set of an approximate level set function using a bulk finite element space.
  - The *diffuse interface* approach [**?**, **?**, **?**] is based on approximating the surface using a phase field type variable and solving a bulk advection diffusion equation with coefficients which are zero or small outside of a transition layer.
- The closest point method for partial differential equations on stationary surfaces, [**?**, **?**, **?**], is based on considering $u(p(x))$ where $p(x) \in \Gamma$ is the point closest to x. The surface partial differential equation is then embedded and discretised in a neighbourhood of Γ using $u(p(x))$. Implementation requires the knowledge or calculation of the closest point $p(x)$. In the cited references this approach has been used to solve a wide variety of equations on stationary surfaces.

1.5. Layout of the paper

The structure of the article is as follows. In Section 2 we illustrate how mesh quality for evolving interpolated hypersurfaces depends on the tangential velocity. The evolving surface finite element method is presented in Section

3. Numerical simulations are described which illustrate how the quality of the finite element mesh on Γ_h influences how accurately PDEs on Γ_h can be approximated. In Section 4 we state two numerical schemes for approximating the evolution of hypersurfaces evolving with forced mean curvature flow. In Section 5 we present numerical simulations of two applications; solid tumour growth and diffusion induced grain boundary motion. In these applications (1.1) holds on hypersurfaces $\Gamma(t)$ that evolve with forced mean curvature flow, where the forcing is a function of the solution of (1.1). The numerical schemes that are used to compute the simulations are obtained by coupling the schemes given in Sections 3 and 4.

All the simulations presented were produced using the finite element toolbox ALBERTA, [**?**] and visualised using the visulaization application PARAVIEW, [**?**].

2. Evolving triangulated surfaces: an example

In this section we illustrate how the quality of the resulting mesh on an interpolating triangulated surface depends on the tangential velocity of the nodes. We suppose that the surface $\Gamma(t), t \in [0,T]$ is known. Given an initial triangulation of the surface at time $t = 0$ we evolve the vertices with a velocity of the form (1.3). It then follows that the vertices always remain on the surface and thus the triangulated surface is an interpolation. There now arises the possibility of evolving the vertices in a direction tangential to the underlying surface by the velocity field arising in the underlying physical model or by an arbitrary tangential velocity which is independent of the PDE model but may be chosen for computational reasons.

As an illustrative example we consider the following axi-symmetric surface (c.f. [**?**]) : let $\varphi(x,t)$ be the prescribed level set function

$$\varphi(x,t) = x_1^2 + x_2^2 + a(t)^2 G(x_3^2/L(t)^2) - a(t)^2$$

and the surface $\Gamma(t)$ be described by the zero level set of φ. A variety of surface shapes may be obtained by varying the prescribed non-negative shape function $G(\cdot)$, $G(0) = 0$ and $G(1) = 1$ and the non-negative time dependent length scales $L(\cdot)$ and $a(\cdot)$ in the, respectively, longitudinal x_3 direction and the radial direction in the (x_1, x_2) plane.

We approximate Γ by an interpolated triangulated surface Γ_h i.e. the triangle vertices lie on $\Gamma(t)$. However we consider two forms for the velocity of the nodes of Γ_h; a velocity that is normal to the surface Γ and a velocity that is both tangential and normal to Γ. We denote the nodes of the triangulation by $\mathbf{X}_i^j(t) = \left(X_i^j(t), Y_i^j(t), Z_i^j(t)\right)$, $i \in \mathcal{I}, j \in \mathcal{J}$ such that $\mathbf{X}_i^j(0) \in \Gamma(0)$ and choose the velocity $\dot{\mathbf{X}}_i^j(t)$ of the nodes such that $\mathbf{X}_i^j(t) \in \Gamma(t)$. For $i \in \mathcal{I}$, given Z_i^0 uniformly distributed in $[-L(0), L(0)]$ with respect to i we choose $Z_i^j(0) = Z_i^0$ and $\left(X_i^j(0), Y_i^j(0)\right)$ to be uniformly distributed with respect to

i on the circle

$$(X_i^j(0))^2 + (Y_i^j(0))^2 = (a(0))^2(1 - G((Z_i^0)^2/(L(0))^2)).$$

Normal motion. We define an interpolation $\Gamma_h(t)$ of $\Gamma(t)$ by evolving the nodes in the normal direction to $\Gamma(t)$ so that

$$\dot{\mathbf{X}}_i^j = V_i \nu_i,\ \nu_i = \frac{\nabla\varphi(\mathbf{X}_i^j, t)}{|\nabla\varphi(\mathbf{X}_i^j, t)|} \text{ and } V_i = -\frac{\varphi_t(\mathbf{X}_i^j, t)}{|\nabla\varphi(\mathbf{X}_i^j, t)|}. \tag{2.1}$$

Normal and tangential motion. Now we choose a particular interpolation $\Gamma_h(t)$ of $\Gamma(t)$ whose nodes move in the tangential as well as the normal direction. We set $a^m = a(t^m)$ and $L^m = L(t^m)$ for $m \in [0, M]$ and set

$$Z_i(t) = Z_i^0 \frac{L(t)}{L^0},\ X_i^j(t) = X_i^j(0)\frac{a(t)}{a(0)},\ Y_i^j(t) = Y_i^j(0)\frac{a(t)}{a(0)} \tag{2.2}$$

so that $\varphi(\mathbf{X}_i^j, t) = 0$.

For certain choices of $G(t)$, $a(t)$ and $L(t)$ moving the nodes of the triangulated surface in the normal direction can result in poor quality meshes compared to the meshes in which there is additional tangential motion of the nodes. An example of this can be seen by taking $G(s) = 200s(s - 199/100)$, $a(t) = 0.1 + 0.05\sin(2\pi t)$ and $L(t) = 1 + 0.2\sin(4\pi t)$. The initial surface is given by the zero level set of the level set function

$$\varphi(x, t) = x_1^2 + x_2^2 + 2x_3^2(x_3^2 - \frac{199}{200}) - 0.01. \tag{2.3}$$

In Figure 1 we show the triangulated surfaces Γ_h^m at $t^m = 0.0, 0.2, 0.4, 0.6$ (with t decreasing vertically) obtained from the two methods described above.

The triangulated surfaces on the left are obtained using the pure normal motion (the location of the nodes were found computationally by solving the ODE system (2.1) numerically) while those on the right are defined by (2.2) and have a non-zero tangential velocity. The top plots show the initial triangulated surface $\Gamma_h(0)$ in which $h = \max_{\sigma_j \in \Gamma_h(0)} \text{diam}(\sigma_j) = 0.0998$. From this figure we see that at $t^m = 0.2$ (second row of plots), the difference between the meshes obtained from the two schemes is quite pronounced. In particular the nodes of the triangulated surface on the right, where tangential motion is present, are quite uniformly distributed over the surface, while the nodes of the triangulated surface on the left are very coarsely separated over some parts of the surface. This coarse triangulation fails to capture the curvature of the surface at these parts, giving rise to a poor approximation to the surface. At $t^m = 0.4$ the meshes obtained from the two schemes are quite similar, with both being roughly uniformly distributed over the surface. At $t^m = 0.6$ again the triangulation on the right produced by (2.2) yields nodes that are uniformly distributed over the surface, while the triangulation on the left, produced by (2.1), comprises of regions in which the triangulation is unnecessarily fine and as a result the remaining the triangulation is coarser than it could be.

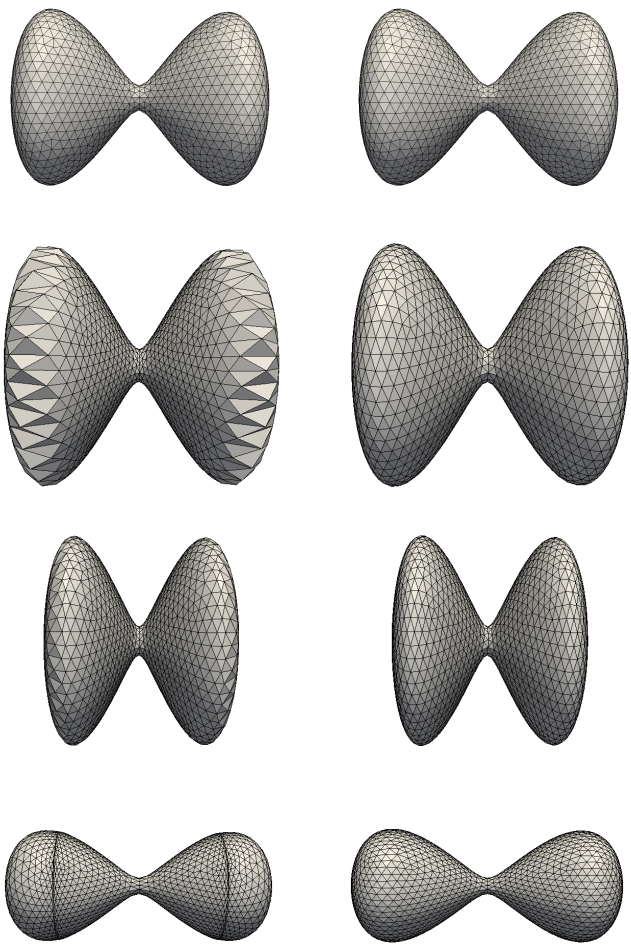

FIGURE 1. Γ_h^m at $t^m = 0.0, 0.2, 0.4, 0.6$ (vertically descending) computed using an approximation to (2.1), left hand plots, and (2.2), right hand plots.

3. Evolving surface finite element method

In this section we discuss the evolving surface finite element method for numerically approximating the solution u of the advection-diffusion equation (1.1).

3.1. Variational formulation

First we define a weak formulation of (1.1). This is obtained by multiplying (1.1) by $\varphi \in H^1(\Gamma(t))$, integrating over $\Gamma(t)$ and integrating by parts to obtain

$$\int_{\Gamma(t)} (\partial^\bullet u\ \varphi + \mathcal{D}\nabla_\Gamma u \cdot \nabla_\Gamma \varphi + u\varphi \nabla_\Gamma \cdot \mathbf{v} - f\varphi) = 0\ \forall \varphi \in H^1(\Gamma(t)). \quad (3.1)$$

Using the transport formula, see [**?**],

$$\frac{d}{dt}\int_{\Gamma(t)} \eta = \int_{\Gamma(t)} \partial^\bullet \eta + \eta\, \nabla_\Gamma \cdot \mathbf{v}$$

we can reformulate (3.1) as

$$\frac{d}{dt}\int_{\Gamma(t)} u\varphi + \int_{\Gamma(t)} (\mathcal{D}\nabla_\Gamma u \cdot \nabla_\Gamma \varphi - f\varphi) = \int_{\Gamma(t)} u\partial^\bullet \varphi\ \forall \varphi \in H^1(\Gamma(t)). \quad (3.2)$$

3.2. Finite element notation

We define $\Gamma_h(t),\ t \in [0,T]$ to be a time dependent triangulated surface such that $\Gamma_h(t) = \cup_{j=1}^J \overline{\sigma_j(t)}$ where $\{\sigma_j(t)\}_{j=1}^J$ is a family of mutually disjoint open triangles. We assume that the triangulated surface $\Gamma_h(t)$ evolves by movement of the triangle vertices in such a way as to maintain the mesh topology for $t \in [0,T]$. We let I be the number and $\mathcal{I}$ the set of vertex indices. For each t we define the finite element space

$$S^h(\Gamma_h(t)) = \{\chi \in C(\Gamma_h(t))|\ \chi|_{\sigma_j} \text{ is piecewise linear for } j = 1 \to J\}.$$

For fully discrete discretizations we set $t^m = m\tau$, $m = 0 \to M$ and for each t^m, $m = 0 \to M$, we define $\Gamma_h^m := \Gamma_h(t^m)$, $\sigma_j^m := \sigma_j(t^m)$ and $\omega^m := \omega(t^m)$. For scalar (and vector) functions $u, v \in L^2(\Gamma_h^m)$ $(u, v \in [L^2(\Gamma_h^m)]^3)$ we introduce the L^2 inner product $\langle \cdot, \cdot \rangle_m$ over Γ_h^m (using the notation of [**?**]):

$$\langle u, v \rangle_m := \int_{\Gamma_h^m} u \cdot v$$

and for piecewise continuous functions u, v we introduce the mass lumped inner product $\langle \cdot, \cdot \rangle_m^h$:

$$\langle u, v \rangle_m^h := \frac{1}{3}\sum_{j=1}^J |\sigma_j^m| \sum_{k=0}^2 (u \cdot v)((\mathbf{q}_{j_k}^m)^-)$$

where $u((\mathbf{q}_{j_k}^m)^-) := \lim_{\sigma_j^m \ni \mathbf{p} \to q_{j_k}^m} u(\mathbf{p})$.

We define a mesh material velocity on $\Gamma_h(t)$ by

$$\mathbf{V}_h^M(x,t) := \sum_{j=1}^N \dot{\mathbf{X}}_j(t)\chi_j(x,t) \text{ for } x \in \Gamma_h(t). \quad (3.3)$$

Also we define an interpolated material velocity on $\Gamma_h(t)$ by

$$\mathbf{V}_h(x,t) := \sum_{j=1}^N v(\mathbf{X}_j(t),t)\chi_j(x,t) \text{ for } x \in \Gamma_h(t) \quad (3.4)$$

which is the interpolation of the material velocity for the PDE.

Given the discrete velocity field $\mathbf{V}_h^M \in (S_h)^{n+1}$ we define *discrete material derivatives* on $\Gamma_h(t)$ element by element through the equations

$$\partial^\bullet_{h,M}\varphi_h|_{E(t)} := (\varphi_{ht} + V_h^M \cdot \nabla\varphi_h)|_{E(t)} \tag{3.5}$$

$$\partial^\bullet_h\varphi_h|_{E(t)} := (\varphi_{ht} + V_h \cdot \nabla\varphi_h)|_{E(t)} \tag{3.6}$$

subsubsection*Transport property of the basis functions

Since the nodal basis functions, $\{\chi_i(\cdot,t)\}_{i=1}^I$ of $S^h(\Gamma_h(t))$ satisfy $\chi_i(\mathbf{X}_j(t) = \delta_{ij}$, the following transport property

$$\chi_{i,t} + \mathbf{V}_h^M \cdot \nabla\chi_i = 0, \text{ for all } i \in \mathcal{I}$$

holds which yields

$$\partial^\bullet_{h,M}\chi_i = 0, \text{ for all } i \in \mathcal{I}. \tag{3.7}$$

3.3. Evolving surface finite element methods

We follow the techniques in [**?**] and derive an evolving surface finite element method for approximating (1.1). In this approach the surface $\Gamma_h(t)$ is an interpolation of $\Gamma(t)$. A continuous in time finite element approximation of (3.2) takes the form:

Find $U_h(\cdot,t) \in S^h(\Gamma_h(t))$ such that

$$\frac{d}{dt}\int_{\Gamma_h(t)} U_h\chi + \int_{\Gamma_h(t)} (\mathcal{D}\nabla_{\Gamma_h}U_h \cdot \nabla_{\Gamma_h}\chi - f_h\chi) = \int_{\Gamma_h(t)} U_h\partial^\bullet_h\chi \ \forall\chi \in S^h(\Gamma_h(t)). \tag{3.8}$$

ESFEM. The first method we consider is the form introduced by Dziuk and Elliott in [**?**] which uses the material velocity in the model for the velocity of the nodes. In this case the material velocity comprises the sum of the normal velocity of the surface and the tangential advection velocity.

Since in this method the vertices of the evolving triangulated surface Γ_h move with the same velocity as the material points on the surface, the nodal basis functions, $\{\chi_i(\cdot,t)\}_{i=1}^I$, of $S^h(\Gamma_h(t))$ satisfy the transport property

$$\chi_{i,t} + \mathbf{V_h} \cdot \nabla\chi_i = 0, \text{ for all } i \in \mathcal{I}$$

which yields

$$\partial^\bullet_h\chi_i = 0, \text{ for all } i \in \mathcal{I}.$$

Thus (3.8) reduces to

$$\frac{d}{dt}\int_{\Gamma_h(t)} U_h\chi_i + \int_{\Gamma_h(t)} (\mathcal{D}\nabla_{\Gamma_h}U_h \cdot \nabla_{\Gamma_h}\chi_i - f_h\chi_i) = 0 \ \forall i \in \mathcal{I}.$$

For a fully discrete approximation we use a fully implicit time discretization; setting U_h^m to represent $U_h(\cdot,t^m)$ we have:

Given Γ_h^{m-1}, Γ_h^m and $U_h^{m-1} \in S^h(\Gamma_h^{m-1})$, find $U_h^m \in S^h(\Gamma_h^m)$ such that for all $i \in \mathcal{I}$

$$\frac{1}{\tau}\langle U_h^m, \chi_i^m\rangle_m^h - \frac{1}{\tau}\langle U_h^{m-1}, \chi_i^{m-1}\rangle_{m-1}^h + \mathcal{D}\langle \nabla_{\Gamma_h^m} U_h^m, \nabla_{\Gamma_h^m}\chi_i^m\rangle_m = \langle f_h^m, \chi_i^m\rangle_m^h. \tag{3.9}$$

ALE ESFEM. The second method is a natural extension of the first in which an arbitrary tangential velocity is added to the normal velocity of the surface. We call this an arbitrary Lagrangian Eulerian evolving surface finite element method (ALE ESFEM).

In this approach the surface $\Gamma_h(t)$ interpolates $\Gamma(t)$ in such a way that the velocity of the vertices may not be the material velocity associated with the PDE, so that $\mathbf{V}_h^M \neq \mathbf{V}_h$. It follows that since

$$\partial_{h,M}^\bullet \chi_i = \chi_{i,t} + \mathbf{V}_h^M \cdot \nabla \chi_i = 0, \text{ for all } i \in \mathcal{I}$$

the nodal basis functions, $\{\chi_i(\cdot,t)\}_{i=1}^I$, of $S^h(\Gamma_h(t))$ satisfy the following transport property

$$\partial_h^\bullet \chi_i = (\mathbf{V}_h - \mathbf{V}_h^M)\cdot \chi_i. \tag{3.10}$$

This leads to the approximation

$$\frac{d}{dt}\int_{\Gamma_h(t)} U_h\chi_i + \int_{\Gamma_h(t)} (\mathcal{D}\nabla_{\Gamma_h} U_h \cdot \nabla_{\Gamma_h}\chi_i) = \int_{\Gamma_h(t)} f_h\chi_i + U_h(\mathbf{v}_h - \mathbf{V}_h)\cdot\nabla_{\Gamma_h}\chi_i \ \forall i \in \mathcal{I}.$$

For a fully discrete approximation we use a fully implicit time discretization to obtain:

Given Γ_h^{m-1}, Γ_h^m and $U_h^{m-1} \in S^h(\Gamma_h^{m-1})$, find $U_h^m \in S^h(\Gamma_h^m)$ such that

$$\begin{aligned}\frac{1}{\tau}\langle U_h^m, \chi_h^m\rangle_m^h \ - \ & \frac{1}{\tau}\langle U_h^{m-1}, \chi_h^{m-1}\rangle_{m-1}^h + \mathcal{D}\langle \nabla_{\Gamma_h^m} U_h^m, \nabla_{\Gamma_h^m}\chi_h^m\rangle_m \\ & + \langle (\mathbf{V}_h^{M,m} - \mathbf{V}_h^m) U_h^m, \nabla_{\Gamma_h^m}\chi_h^m\rangle_m^h \\ & - \langle f_h^m, \chi_h^m\rangle_m^h = 0 \ \forall \chi_h^m \in S^h(\Gamma_h^m).\end{aligned} \tag{3.11}$$

Here $\mathbf{V}_h^{M,m} := \sum_{i=1}^I \dot{\mathbf{X}}_i^m \chi_i^m$ and $\mathbf{V}_h^m = \sum_{i=1}^I \mathbf{v}(\mathbf{X}_i^m)\chi_i^m$.

Remark 3.1. If $\dot{\mathbf{X}}_i^m = \mathbf{v}(\mathbf{X}_i^m)$ then (3.11) reduces to (3.9).

Remark 3.2. The key difference between these two methods is that in the ESFEM method the normal velocity of the surface and the advective velocity do not explicitly appear in the discretization. The numerical method requires only knowledge of the position of the vertices. In ALE ESFEM an advective velocity term appears in the discretization.

3.4. A numerical example comparing the schemes

We now investigate how the quality of the triangulated surface effects the accuracy with which a PDE can be solved on it. In particular we use the two triangulated surfaces (Figure 1) obtained using (2.1) and (2.2) presented in Section 2 on which to approximate (1.1). In the example we choose $u(x,t) = e^{-6t}x_1x_2$ to be the solution of (1.1) with f computed from the equation

$$f = \partial^\bullet u \ - \mathcal{D}\Delta_\Gamma u + u\nabla_\Gamma \cdot \mathbf{v}$$

with $\mathbf{v} = V\boldsymbol{\nu}$.

We present the results in Figure 2 in which we plot the absolute error $|U_h^m - u|$ on the triangulated surfaces Γ_h^m displayed in Figure 1. We see that the maximum errors of $|U_h^m - u|$ plotted on the triangulated surfaces on the right, which had uniformly distributed meshes, are consistently less than those on the left where the meshes were of a lower quality, with some very coarse elements in places and also some unnecessarily fine ones.

4. Curvature dependent interface motion

In this section we study two numerical approaches for approximating an interface $\Gamma(t)$ that evolves with forced mean curvature flow, such that

$$V = \kappa + \alpha g. \tag{4.1}$$

Here κ is the mean curvature of $\Gamma(t)$ (with the convention that $\kappa < 0$ for a sphere oriented so that the normal is pointing outwards) and $g(x,t)$ is a given forcing function. The first approach was introduced by Dziuk in [**?**], while the second is the method of Barrett, Garcke and Nürnberg presented in [**?**, **?**]. In Dziuk's approach the nodes of the triangulated surface move with the same velocity as the material points, i.e. in a direction normal to the surface, while in the model of Barrett, Garcke and Nürnberg the nodes of the triangulated surface move with an additional non-physical tangential velocity.

In order to discretize (4.1) we first write it in a parametric formulation: for parametrizations $\mathbf{x} : \Sigma \times [0,T] \to \mathbb{R}^3$, with $\mathbf{x}(\cdot,0) = \mathbf{x}_0(\cdot)$, where Σ is a suitable compact reference manifold in $\mathbb{R}^3$. Using the identity

$$\kappa\boldsymbol{\nu} = \Delta_\Gamma \mathbf{x} \tag{4.2}$$

in which the sign convention fixes the orientation of the normal, (4.1) can be written in the forms

$$\mathbf{x}_t = \Delta_\Gamma \mathbf{x} + \alpha g\boldsymbol{\nu} \tag{4.3}$$

$$V := \mathbf{x}_t \cdot \boldsymbol{\nu} = \kappa + \alpha g. \tag{4.4}$$

The identity in (4.2) was used for the first time by Dziuk in [**?**] in designing a finite element method for mean curvature flow. Note that a parameterization which satisfies (4.3) yields a motion for material points on the surface which is purely normal in direction whereas a parameterization which satisfies (4.4) may also have tangential motion.

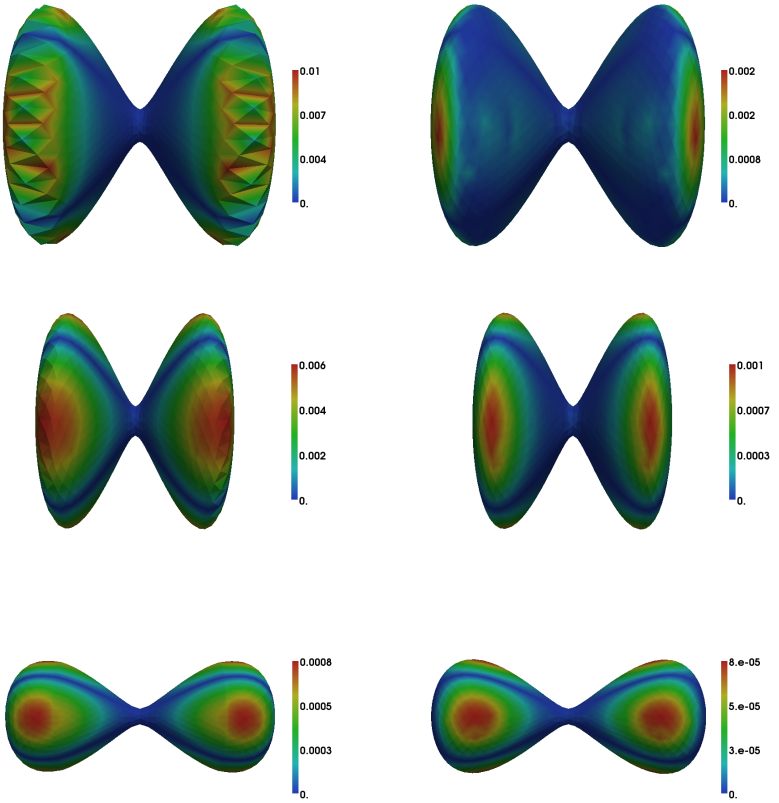

FIGURE 2. $|U_h^m - u|$, $t^m = 0.2, 0.4, 0.6$, plotted on the triangulated surfaces displayed in Figure 1.

4.1. More finite element notation

We denote the vertices of $\sigma_j(t)$ by $\{\mathbf{q}_{j_k}(t)\}_{k=0}^2$ and we define the unit normal $\boldsymbol{\nu}_h(t)$ to $\Gamma_h(t)$ such that

$$\boldsymbol{\nu}_j(t) := \boldsymbol{\nu}_h|_{\sigma_i(t)} := \frac{(\mathbf{q}_{j_1}(t) - \mathbf{q}_{j_0}(t)) \times (\mathbf{q}_{j_2}(t) - \mathbf{q}_{j_0}(t))}{|(\mathbf{q}_{j_1}(t) - \mathbf{q}_{j_0}(t)) \times (\mathbf{q}_{j_2}(t) - \mathbf{q}_{j_0}(t))|} \text{ for } j = 1 \to J.$$

We set

$$\underline{S}^h(\Gamma_h(t)) := \{\chi \in [S^h(\Gamma_h(t))]^3 | \}.$$

Next we follow the authors in [**?**] and introduce a weighted normal, $\omega(t) := \sum_{i=1}^I \omega_i(t)\chi_i$ such that $\omega_i(t)$ can be interpreted as a weighted normal defined

at the node $\mathbf{q}_i(t)$ of the surface $\Gamma_h(t)$ and is defined by

$$\omega_i(t) := \frac{1}{|\Lambda_i(t)|} \sum_{\sigma_j(t)\in\mathcal{T}_h(t)} |\sigma_j(t)|\boldsymbol{\nu}_j(t) \tag{4.5}$$

where $\mathcal{T}_h(t) := \{\sigma_j(t) : \mathbf{q}_i(t) \in \overline{\sigma_j(t)}\}$, $\Lambda_i(t) := \cup_{\sigma_j(t)\in\mathcal{T}_h(t)}\overline{\sigma_j(t)}$ and $|\sigma_j(t)|$ is the measure of $\sigma_j(t)$.

4.2. Discretizations of forced mean curvature flow

We now give two finite element approximations of (4.1).

Scheme A

Following Dziuk in [**?**] the velocity law (4.3) may be approximated by the following scheme:

Given a parametrization $\mathbf{X}^{m-1} \in \underline{S}^h(\Gamma_h^{m-1})$ of Γ_h^{m-1} find $\mathbf{X}^m \in \underline{S}(\Gamma_h^{m-1})$ such that

$$\frac{1}{\tau}\langle \mathbf{X}^m - \mathbf{X}^{m-1}, \chi\rangle^h_{m-1} + \langle \nabla_{\Gamma_h^{m-1}}\mathbf{X}^m, \nabla_{\Gamma_h^{m-1}}\chi\rangle_{m-1} = \alpha\langle (g_h^{m-1}\boldsymbol{\nu}_h{}^{m-1}, \chi\rangle^h_{m-1} \ \forall \chi \in \underline{S}^h(\Gamma_h^{m-1}). \tag{4.6}$$

Scheme B

Following the ideas of Barrett, Garcke and Nürnberg in [**?**], (4.4) may be approximated by:

Given a parametrization $\mathbf{X}^{m-1} \in \underline{S}(\Gamma_h^{m-1})$ of Γ_h^{m-1} find $\{\mathbf{X}^m, \kappa^m\} \in \underline{S}(\Gamma_h^{m-1}) \times S^h(\Gamma_h^{m-1})$ such that

$$\frac{1}{\tau}\langle \mathbf{X}^m - \mathbf{X}^{m-1}, \chi\boldsymbol{\nu}_h{}^{m-1}\rangle^h_{m-1} - \langle \kappa^m, \chi\rangle^h_{m-1} = \alpha\langle g_h^{m-1}, \chi\rangle^h_{m-1} \ \forall \chi \in S^h(\Gamma_h^{m-1}) \tag{4.7a}$$

$$\langle \kappa^m\boldsymbol{\nu}_h{}^m, \chi\rangle^h_{m-1} + \langle \nabla_{\Gamma_h^{m-1}}\mathbf{X}^m, \nabla_{\Gamma_h^{m-1}}\chi\rangle_{m-1} = 0 \ \forall \chi \in \underline{S}(\Gamma_h^{m-1}). \tag{4.7b}$$

4.3. Comparison of how well Scheme A and Scheme B approximate (4.1)

We investigate how well the two schemes (4.6) and (4.7a,b) approximate the geometric evolution (4.1). In particular we choose $g(x,t)$ so that $\Gamma(t)$ evolves according to a prescribed law. We study the evolution of the closed surface given by the zero level set of the level set function

$$\varphi(x,t) = \frac{x_1^4}{1+16\sin(t)} + x_2^4 + x_3^4 - 1. \tag{4.8}$$

The initial surface $x_1^4 + x_2^4 + x_3^4 = 1$ is displayed in Figure 3 (left hand plot) together with the initial triangulated surface $\Gamma_h(0)$ (right hand plot) in which $h = \max_{\sigma_j\in\Gamma_h(0)} \mathrm{diam}(\sigma_j) = 0.215$. Since the surface $\Gamma(t)$ is defined by the zero level set of $\varphi(x,t)$, the accuracy of the two schemes can be gauged by the error $|\varphi(\mathbf{X}_i^m, t^m)|$, for $i \in \mathcal{I}$ and $t^m \in (0,T)$. In Figure 4 we display plots of $|\varphi(\mathbf{X}_i^m, t^m)|$, for $t^m = 0.5, 1.0, 1.5$, on the triangulated surfaces Γ_h^m obtained using Scheme A (left hand plots) and Scheme B (right hand plots). From this figure we see that at the final time $t^m = 1.5$ the meshes produced by

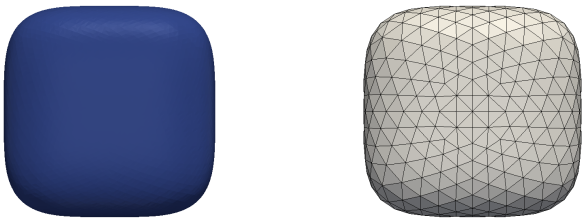

FIGURE 3. Zero level set of (4.8) and initial triangulated surface $\Gamma_h(0)$.

	τ	C1 cube	C2 sphere
Mesh 1	$\tau = 1.0 \cdot 10^{-2}$	$h = 2.15 \cdot 10^{-1}$	$h = 1.10 \cdot 10^{-1}$
Mesh 2	$\tau = 5.0 \cdot 10^{-3}$	$h = 1.89 \cdot 10^{-1}$	$h = 5.50 \cdot 10^{-2}$
Mesh 3	$\tau = 2.5 \cdot 10^{-3}$	$h = 9.64 \cdot 10^{-2}$	$h = 2.75 \cdot 10^{-2}$
Mesh 4	$\tau = 1.25 \cdot 10^{-3}$	$h = 4.86 \cdot 10^{-2}$	$h = 1.38 \cdot 10^{-2}$

TABLE 1. Timesteps and initial mesh sizes (to 2 decimal places).

the two schemes are quite different; the mesh produced by Scheme A is finer in the centre of the domain and coarser towards the corners while the mesh produced by Scheme B is more uniform. These features can also be seen at the earlier times $t = 0.5$ and $t = 1.0$ but they are less pronounced. Furthermore from the values of $|\varphi(\mathbf{X}_i^m, t^m)|$ that are displayed on the surfaces we can see that triangulated surface produced by Scheme B yields a significantly better approximation to the surface $\Gamma(t)$ than the triangulated surface produced by Scheme A.

In order to assess the convergence of the two schemes we solve the above problem using four initial meshes and four timestep sizes, see Table 1, Column C1. We display the results in Figure 5 in which the error

$$e(t^m) = \max_{i \in \mathcal{I}} |\varphi(\mathbf{X}_i^m, t^m)|, \text{ for } t^m \in (0, T)$$

is plotted against time, for the two schemes. The results from Scheme A are plotted with a solid blue line and the results for Scheme B are plotted with a dashed red line. In each figure there are four plots, showing results from Meshes $1 - 4$ (left hand plot to right hand plot). From these results we see evidence of linear convergence in both schemes, also we see that the difference in the errors between the two schemes is quite pronounced as the errors for Scheme B are effectively half those of Scheme A.

4.4. Solving PDEs on the triangulations arising from Schemes A and B

We now investigate how the quality of the triangulated surface effects the accuracy of the solution of (1.1). We use (3.11) to approximate (1.1) on the evolving hypersurfaces Γ_h produced by Schemes A and B. In particular

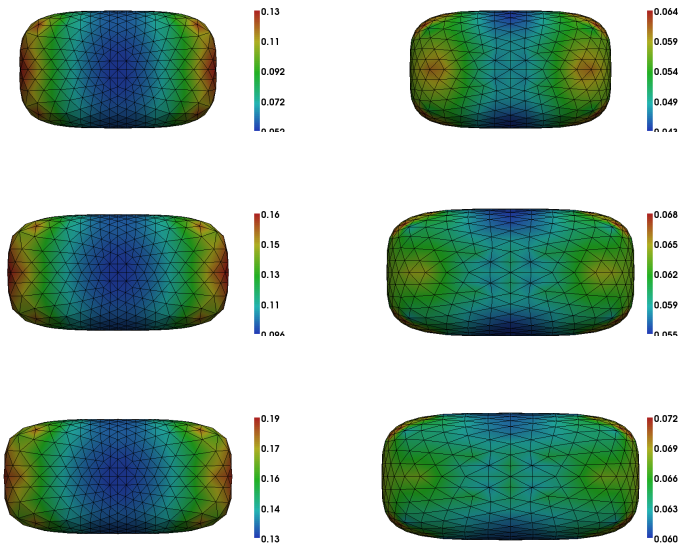

FIGURE 4. $|\varphi(\mathbf{X}_i^m, t^m)|$, $t^m = 0.5, 1.0, 1.5$ (vertically descending) plotted on triangulated surfaces Γ_h^m produced using Scheme A (left) and Scheme B (right).

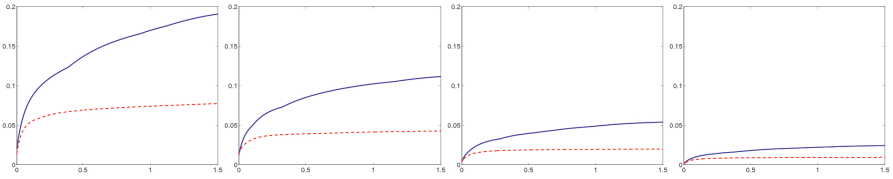

FIGURE 5. Plots of $e(t) = \max_{i\in\mathcal{I}} |\varphi(\mathbf{X}_i^m, t)|$, for (4.8) on Meshes $1-4$, with Scheme A solid blue line and Scheme B dashed red line.

we note that when using Scheme A the velocity of the nodes of Γ_h is the same as the velocity of the material points, i.e. $\dot{\mathbf{X}}_i^m = \mathbf{v}(\mathbf{X}_i^m)$ and hence for Scheme A (3.9) can be used to approximate (1.1). In the example we choose $u(x,t) = e^{-6t}x_1x_2$ to be the solution of (1.1) with f computed from the equation

$$f = \partial^\bullet u \ - \mathcal{D}\Delta_\Gamma u + u\nabla_\Gamma \cdot \mathbf{v}$$

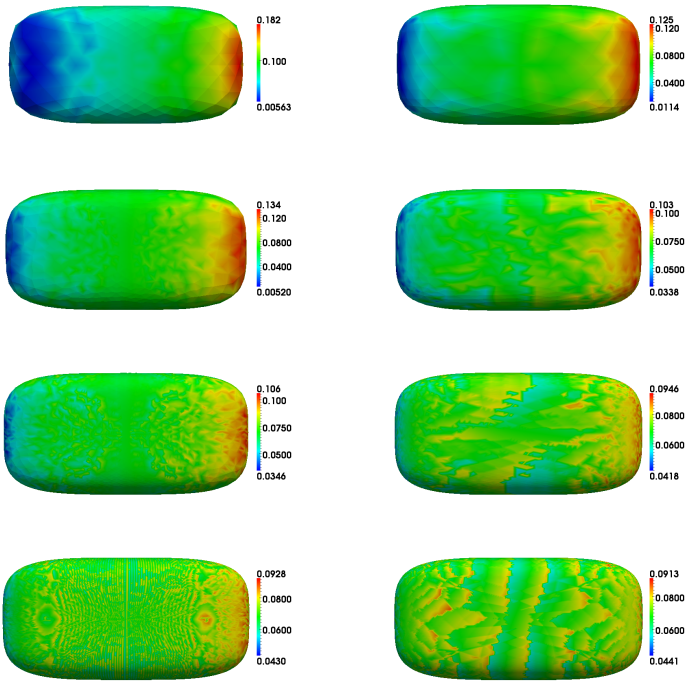

FIGURE 6. $|u_{err}(\mathbf{X}_i^m, t^m)|$ at $t^m = 1.5$ plotted on triangulated surfaces Γ_h^m for Meshes 1 - 4 (vertically decreasing) produced using Scheme A (left) and Scheme B (right).

with $\mathbf{v} = V\boldsymbol{\nu}$.

It is not obvious how to compute errors between u and U_h since $u(x, t^m) \in \Gamma(t^m)$ and $U_h^m \in \Gamma_h(t^m)$ and the nodes of $\Gamma_h(t^m)$ do not necessarily lie on $\Gamma(t^m)$. This is in contrast to Section 3.3 in which the nodes of the approximated surfaces Γ_h do lie on Γ.

To this end we consider the error

$$u_{err}(\mathbf{X}_i^m) := |U_h(\mathbf{X}_i^m) - u(p(\mathbf{X}_i^m), t^m)|$$

where $p(\mathbf{X}_i^m, t^m)$ is the point on $\tilde{\Gamma}_h(t^m)$ that is closest to $\mathbf{X}_i^m$ and $\tilde{\Gamma}_h(t^m)$ is an approximation to $\Gamma(t^m)$. In particular $\tilde{\Gamma}_h(t^m)$ is a triangulated surface whose nodes lie on $\Gamma(t^m)$. In the example we have 50690 evenly distributed nodes on $\tilde{\Gamma}_h(t^m)$.

In Figure 6 we display plots of $u_{err}(\mathbf{X}_i^m, t^m)$ on the surfaces produced by Scheme A (left) and Scheme B (right) at $t = 1.5$ using the initial mesh and timestep of Meshes 1 - 4 (vertically decreasing) displayed in Table 1,

Column C1. From these results we see that for errors are consistently lower when (3.11) is coupled to Scheme B than they are when (3.9) is coupled to Scheme A. However as the mesh size and timestep are reduced both schemes for evolving the triangulated surface yield good meshes and as a consequence the magnitude of the errors become similar for Mesh 4.

5. Applications coupling surface PDEs to surface evolution

In this section we consider two applications for which the mathematical problem is to determine a scalar field u and hypersurfaces $\Gamma(t) \subset \mathbb{R}^3$ such that u solves

$$\partial^\bullet u \ - \mathcal{D}\Delta_\Gamma u + u\nabla_\Gamma \cdot \mathbf{v} = f(V,u) \text{ on } \Gamma(t) \tag{5.1a}$$

with $\mathbf{v} = V\boldsymbol{\nu}$ and $\Gamma(t)$ evolves with the velocity law

$$V = \kappa + g(u). \tag{5.1b}$$

Here we note that the system (5.1a,b) is strongly coupled as the evolution of $\Gamma(t)$ is partially determined by the solution of the PDE that holds on $\Gamma(t)$. Also note that in these examples there is no surface tangential advective velocity and we take the material velocity to be the normal velocity of the surface. The first application we consider is an example in mathematical biology, namely the growth of solid tumours, while the second is an example in material science, namely diffusion induced grain boundary motion (DIGM).

Before introducing these models we couple the schemes presented in Section 3 and 4 to obtain two approximations of (5.1a,b).

5.1. Discretizations of the coupled system (5.1a,b)

The velocity law (5.1b) gives rise to hypersurfaces evolving in the normal direction and by using Scheme A from Section 4 we can ensure that the nodes of the triangulated surface also move in the normal direction and hence we may use (3.9) to approximate (5.1a). This results in the following discretization:

Scheme $\tilde{\mathbf{A}}$

Given a parametrization $\mathbf{X}^{m-1} \in \underline{S}^h(\Gamma_h^{m-1})$ of Γ_h^{m-1} and an approximation $U_h^{m-1} \in S^h(\Gamma_h^{m-1})$ to $u(t_{m-1})$, find $\mathbf{X}^m \in \underline{S}^h(\Gamma_h^{m-1})$ such that for all $\chi \in \underline{S}^h(\Gamma_h^{m-1})$ we have

$$\frac{1}{\tau}\langle \mathbf{X}^m - \mathbf{X}^{m-1}, \chi\rangle_{m-1}^h + \langle \nabla_{\Gamma_h^{m-1}} \mathbf{X}^m, \nabla_{\Gamma_h^{m-1}} \chi\rangle_{m-1} = \alpha \langle (g(U_h^{m-1})\boldsymbol{\nu}_h{}^{m-1}, \chi\rangle_{m-1}^h.$$

Now find $U_h^m \in S^h(\Gamma_h^m)$ such that $\forall i \in \mathcal{I}$

$$\begin{aligned}\frac{1}{\tau}\langle U_h^m, \chi_i^m\rangle_m^h + \mathcal{D}\langle \nabla_{\Gamma_h^m} U_h^m, \nabla_{\Gamma_h^m}\chi_i^m\rangle_m \\ = \langle f_h(V_h^m, U_h^m, U_h^{m-1}), \chi_i^m\rangle_m^h + \frac{1}{\tau}\langle U_h^{m-1}, \chi_i^{m-1}\rangle_{m-1}^h.\end{aligned}$$

Here $f_h(V_h^m, U_h^m, U_h^{m-1})$ is an approximation of $f(V_h^m, U_h^m)$ that it is linear in U_h^m.

Using Scheme B from Section 4 to approximate (5.1b) results in the possibility of the nodes of the triangulated surface moving in a direction tangential to $\Gamma_h(t)$, such that $\mathbf{V}_h^M = V_h\boldsymbol{\nu}_h + \mathbf{T}_h$. Since the velocity of the material points on Γ_h is $\mathbf{V}_h = V_h\boldsymbol{\nu}_h$, when approximating (5.1a) we use the ALE ESFEM approximation (3.11) with $\mathbf{V}_h^{M,m} - \mathbf{V}_h^m = \mathbf{T}_h^m$. Thus an alternative approximation to the system (5.1a,b) takes the following form:

Scheme $\tilde{\mathbf{B}}$

Given a parametrization $\mathbf{X}^{m-1} \in \underline{S}^h(\Gamma_h^{m-1})$ of Γ_h^{m-1} and an approximation $U_h^{m-1} \in S^h(\Gamma_h^{m-1})$ to $u(t_{m-1})$, find $\{\mathbf{X}^m, \kappa^m\} \in \underline{S}^h(\Gamma_h^{m-1}) \times S^h(\Gamma_h^{m-1})$ such that

$$\frac{1}{\tau}\langle \mathbf{X}^m - \mathbf{X}^{m-1}, \chi \boldsymbol{\nu}_h{}^{m-1}\rangle_{m-1}^h - \langle \kappa^m, \chi\rangle_{m-1}^h = \alpha\langle g(U_h^{m-1}), \chi\rangle_{m-1}^h \;\forall \chi \in S^h(\Gamma_h^{m-1})$$

$$\langle \kappa^m \boldsymbol{\nu}_h{}^m, \chi\rangle_{m-1}^h + \langle \nabla_{\Gamma_h^{m-1}} \mathbf{X}^m, \nabla_{\Gamma_h^{m-1}} \chi\rangle_{m-1} = 0 \;\forall \chi \in \underline{S}^h(\Gamma_h^{m-1}).$$

Now find $U_h^m \in S^h(\Gamma_h^m)$ such that $\forall \chi_h^m \in S^h(\Gamma_h^m)$,

$$\frac{1}{\tau}\langle U_h^m, \chi_h^m\rangle_m^h + \mathcal{D}\langle \nabla_{\Gamma_h^m} U_h^m, \nabla_{\Gamma_h^m}\chi_h^m\rangle_m + \langle U_h^m, \mathbf{T}_h^m \cdot \nabla_{\Gamma_h^m}\chi^m\rangle_m^h = \langle f_h(V_h^m, U_h^m, U_h^{m-1}), \chi_h^m\rangle_m^h + \frac{1}{\tau}\langle U_h^{m-1}, \chi_h^{m-1}\rangle_{m-1}^h.$$

Here

$$\mathbf{T}_h^m = \sum_{i=1}^{I} \mathbf{T}_i^m \chi_i^m$$

with

$$\mathbf{T}_i^m := \frac{1}{\tau}\left([\mathbf{X}_i^m - \mathbf{X}_i^{m-1}] - [\mathbf{X}_i^m - \mathbf{X}_i^{m-1}]\cdot\omega_i^m\omega_i^m\right).$$

In the following two subsections we use Schemes $\tilde{\mathrm{A}}$ and $\tilde{\mathrm{B}}$ to obtain numerical simulations of solid tumour growth and diffusion induced grain boundary motion.

5.2. Solid tumour growth

In [**?**] the authors present a model for the growth of solid tumours during their avascular growth phase. Although the tumour itself is a solid object while it is in its avascular growth phase, the cells towards the centre of the tumour are deprived of essential nutrients and eventually die, resulting in the profliferating cells lying essentially only on the surface of the tumour. In this model the development of a hetrogeneous chemical pre-pattern on the surface of the tumour is considered. Here the pre-pattern predisposes cells in regions where the concentration of the growth-promoting factor is high to invasion leading to invasive growth. Thus the evolution of the tumour is partially determined by the chemical pre-pattern on its surface. The resulting mathematical problem is that of determining a surface $\Gamma(t) \subset \mathbb{R}^3$ and

scalar functions $u(x,t)$ and $w(x,t)$, that respectively represent the growth-promoting and growth-inhibiting factors, such that

$$\partial^\bullet u = \Delta_\Gamma u - u\nabla_\Gamma \cdot \mathbf{v} + \gamma f_1(u,w) \text{ on } \Gamma(t) \tag{5.2a}$$

$$\partial^\bullet w = D_c\Delta_\Gamma w - w\nabla_\Gamma \cdot \mathbf{v} + f_2(u,w) \text{ on } \Gamma(t) \tag{5.2b}$$

with $\mathbf{v} = V\boldsymbol{\nu}$ and

$$V = \varepsilon\kappa + \delta u \ \forall t \geq \bar{t}. \tag{5.2c}$$

In the velocity law (5.2c), $\bar{t}$ denotes the transition time from the avascular phase to the vascular phase, the second term on the right comes from the suggestion in [**?**] that growth should be faster in regions of higher concentration of the growth-promoting factor u and the first term on the right is a regularising term to obtain smooth motion. Lastly δ and ε are positive constants. In the surface partial differential equations (5.2a,c) D_c is the diffusion coefficient ratio and f_1 and f_2 model the interactions between the two chemicals. In particular consider the activator-depleted substrate model, [**?**], also known as the Brusselator model, in which

$$f_1(u,w) = \gamma(a - u + u^2 w) \text{ and } f_2(u,w) = \gamma(b - u^2 w). \tag{5.2d}$$

Here γ, a and b are positive constants with $\sqrt{\gamma}$ being proportional to the area of the domain and γ can be seen as the relative strength of the interaction and diffusion terms.

We approximate (5.2a-c) using the obvious extension of Schemes $\tilde{\text{A}}$ and $\tilde{\text{B}}$ to deal with (5.2b). In these schemes we set

$$f_{h1}(U_h^m, U_h^{m-1}, W_h^{m-1}) = \gamma(a - U_h^m + U_h^m U_h^{m-1} W_h^m)$$

and

$$f_{h2}(U_h^m, W_h^m) = \gamma(b - (U_h^m)^2 W_h^m).$$

Remark 5.1. In [**?**] a model for tumour growth is presented together with a finite element approximation of the form of Scheme $\tilde{\text{A}}$.

5.2.1. Numerical simulations of tumour growth. In Figures 7 and 8 we show a simulation of the tumour growth model. For initial data we took $\Gamma(0)$ to be the unit sphere and we set U_h^0 to be the distribution seen in the left hand plot of Figure 7. We took $a = 0.1$ and $b = 0.9$, $D_c = 10$, $\gamma = 100$, $\varepsilon = 0.01$, $\delta = 0.4$ and $\bar{t} = 0$. In Figure 7 the approximate solutions $U_h(t^m)$ obtained from Scheme $\tilde{\text{A}}$ using Mesh 3 (see Table 1 Column C2) are plotted on $\Gamma_h(t^m)$ at times $t^m = 0, 1.2, 1.8$.

We computed this simulation on Meshes 1 - 4, see Table 1 Column C2, for both Scheme $\tilde{\text{A}}$ and Scheme $\tilde{\text{B}}$. For the early stages of motion $t \in (0, 0.6)$ the two schemes gave graphically indistinguishable results. However for later stages of motion on the coarser meshes this was not the case. This can be seen in Figure 8 where the triangulated surface Γ_h^m, at $t^m = 1.4$, and the approximate solution $U_h(t^m)$, are displayed for both schemes using Mesh 1. The mesh obtained from Scheme $\tilde{\text{A}}$ is shown in the second plot and the mesh from Scheme $\tilde{\text{B}}$ in the fourth plot. Here we see that the artificial

tangential motion produced in Scheme $\tilde{\text{B}}$ results in the triangulated surface becoming distorted, while the triangulated surface arising from Scheme $\tilde{\text{A}}$ is quite uniform. In the first and third plots of Figure 8, we see how these different meshes gives rise to different values of U_h. In particular we see the substantial difference between the approximate solution U_h^m obtained from Scheme $\tilde{\text{A}}$ and the approximate solution U_h^m obtained from Scheme $\tilde{\text{B}}$.

We note that the distorted mesh effect given by Scheme $\tilde{\text{B}}$ only occurs for Mesh 1, when the finer meshes, Meshes 2 - 4, are used the mesh remains uniform and the difference between the solutions U_h of the two schemes reduces substantially.

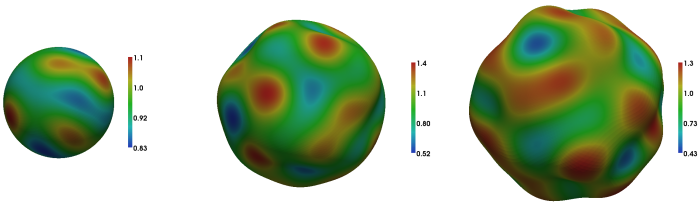

FIGURE 7. An example of tumour growth using Scheme $\tilde{\text{A}}$ with Mesh 3: U_h plotted on Γ_h^m for $t^m = 0,\ 1.2,\ 1.8$.

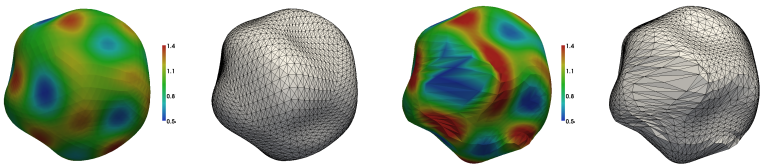

FIGURE 8. An example of tumour growth using Mesh 1: U_h plotted on Γ_h^m together with the triangulated surface Γ_h^m, $t^m = 1.4$. Scheme $\tilde{\text{A}}$, (first and second plots), Scheme $\tilde{\text{B}}$ (third and fourth plots).

5.3. Diffusion induced grain boundary motion

The physical phenomenon of diffusion induced grain boundary motion occurs when grain boundaries in thin metallic films are induced to move due to the absorption of solute from an external vapour, [**?**, **?**]. The mathematical problem is that of determining a surface $\Gamma(t) \subset \mathbb{R}^3$ and a scalar function u such that

$$\partial^\bullet u \ - \mathcal{D}\Delta_\Gamma u + u\nabla_\Gamma \cdot \mathbf{v} = -\beta V u \text{ on } \Gamma(t) \tag{5.3a}$$

with $\mathbf{v} = V\boldsymbol{\nu}$ and

$$V = \kappa + \alpha u^2. \tag{5.3b}$$

	τ	C1 growing cylinder	C2 shrinking cylinder	C3 plane
Mesh 1	$\tau = 1.0 \cdot 10^{-2}$	$h = 3.28 \cdot 10^{-2}$	$h = 1.38 \cdot 10^{-1}$	$h = 3.13 \cdot 10^{-2}$
Mesh 2	$\tau = 5.0 \cdot 10^{-3}$	$h = 1.64 \cdot 10^{-2}$	$h = 6.93 \cdot 10^{-2}$	$h = 1.56 \cdot 10^{-2}$
Mesh 3	$\tau = 2.5 \cdot 10^{-3}$	$h = 8.20 \cdot 10^{-3}$	$h = 3.48 \cdot 10^{-2}$	$h = 7.81 \cdot 10^{-3}$
Mesh 4	$\tau = 1.25 \cdot 10^{-3}$	$h = 4.10 \cdot 10^{-3}$	$h = 1.53 \cdot 10^{-2}$	$h = 3.91 \cdot 10^{-3}$

TABLE 2. Timesteps and initial mesh sizes (to 2 decimal places).

Here the surface $\Gamma(t)$ represents the grain boundary, $u(x,t)$ is a scalar function that denotes the concentration of solute on $\Gamma(t)$ and α and β are positive physical constants. The free boundary problem (5.3a,b) arises from formal asymptotics on the phase field model for diffusion induced grain boundary motion presented in [**?**] and existence and uniqueness of classical solutions to this free boundary problem are presented in [**?**]. Mathematical and numerical analysis of phase field and parametric models for diffusion induced grain boundary motion can be found in [**?**, **?**, **?**, **?**, **?**].

When studying this model we consider the geometrical configuration of a domain $\Omega := (-1,1) \times (0,L) \times (-1,1)$ that contains a single hypersurface $\Gamma(t)$. Here Ω represents the thin film in which the grain boundary is contained. We concentrate on two configurations; in the first $\Gamma(t)$ spans the height (x_3 direction) and width (x_1 direction) of the domain, while in the second $\Gamma(t)$ spans the height of the domain and a horizontal cross section of $\Gamma(t)$ yields a closed curve in the plane $x_3 =$ constant. For example $\Gamma(t)$ could be a cylinder with axis perpendicular to the x_3 axis. In both configurations we assume that $\Gamma(t)$ never comes in contact with the planes $x_2 = 0$ or $x_2 = L$.

We supplement (5.3a) with the following boundary data and initial data

$$u(x,t) = u_+ \ \forall x \in \Gamma(t) \cap \{x_3 = 1\},$$
$$u(x,t) = u_- \ \forall x \in \Gamma(t) \cap \{x_3 = -1\}, \tag{5.3c}$$

where u_+ and u_- are positive constants, and we impose the natural boundary condition

$$\nabla u \cdot \boldsymbol{\mu} = 0 \ \forall x \in \Gamma(t) \cap \{x_1 = \pm 1\}, \tag{5.3d}$$

where $\boldsymbol{\mu}$ is tangential to $\partial\Gamma(t)$ and normal to $\partial\Omega$. Furthermore we set

$$u(x,0) = 0 \ \forall x \in \Gamma(0). \tag{5.3e}$$

Physically the conditions (5.3c-e) imply that solute is only absorbed into the grain boundary from the top and bottom of the film and that initially there is no solute in the film.

To supplement (5.3b) we set $\Gamma(0)$ to be attached orthogonally to the boundaries $x_1 = \pm 1$ and $x_3 = \pm 1$ of Ω and we impose that $\Gamma(t)$ remains orthogonally attached to these boundaries for all t.

In this application $\Gamma(t)$ is not a closed surface so in order to obtain finite element approximations of (5.3a-e) we need to adapt Schemes $\tilde{\text{A}}$ and $\tilde{\text{B}}$ to incorporate the techniques introduced in [**?**, **?**] to deal with the boundary conditions. Also we approximate the fourth term in (5.3a) by $\beta V_h^m U_h^m$ where $V_h^m = \sum_{i=1}^I V_i^m \chi_i^m$ with $V_i^m := \frac{1}{\tau}\left([\mathbf{X}_i^m - \mathbf{X}_i^{m-1}]\right) \cdot \omega_i^m$.

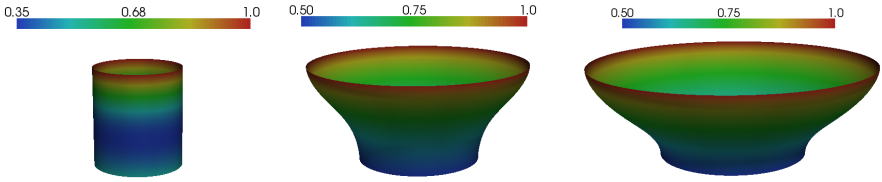

FIGURE 9. Simulation 1: U_h^m plotted on Γ_h^m at $t^m = 0.0, 0.16, 0.26$, (Scheme $\tilde{\text{B}}$).

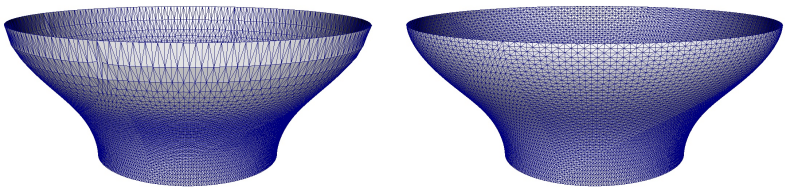

FIGURE 10. Simulation 1: Comparison of the triangulated surfaces Γ_h^m at $t^m = 0.26$, (Scheme $\tilde{\text{A}}$) (left) and (Scheme $\tilde{\text{B}}$) (right).

5.3.1. Numerical simulations of diffusion induced grain boundary motion. We show three numerical simulations obtained using suitable adaptations of Schemes $\tilde{\text{A}}$ and $\tilde{\text{B}}$.

Simulation 1

We set $\Omega = (-1,1) \times (0,2) \times (-1,1)$ and $\Gamma(0)$ to be the cylindrical surface $x_1^2 + x_2^2 = 0.81$, $-1 \leq x_3 \leq 1$, with ν being the outward pointing normal such that on $\Gamma_h(0)$ we have $\kappa < 0$. The value of $U^0(x)$ can be seen in the left hand plot of Figure 9, in particular we took $U^0(x)$ such that $V(0) = \kappa(0) + \alpha(U^0)^2 > 0$. We set $u_+ = 1$, $u_- = 0.5$, $\mathcal{D} = 10$, $\alpha = 12$, $\beta = 1$. In Figures 9 - 12 we show results for Simulation 1. The first figure, Figure 9, contains results obtained using Scheme $\tilde{\text{B}}$, it displays the approximate solution $U_h(t^m)$ on $\Gamma_h(t^m)$ at times $t^m = 0.0, 0.16, 0.26$. Here we are in the situation where the initial concentration is chosen so that it dominates over the negative curvature and the cylinder expands. Initially the curvature is constant on the domain and hence from (5.3b) we see that the value of the concentration will determine which parts of the interface move the most. Since the initial concentration is greater at the top of the domain than at the bottom (left hand plot) we see that the top parts of

the interface move the quickest. Also the external concentration is chosen so that $u_+ = 2u_-$ and hence this concentration profile continues through the simulation. Furthermore as the interface expands the the curvature becomes smaller in absolute value and thus contributes to the normal velocity of Γ_h increasing as t increases.

We computed the problem on Meshes 1 - 4, see Table 2 Column C1, for both Scheme $\tilde{\text{A}}$ and Scheme $\tilde{\text{B}}$. For all of the meshes in the early stages of evolution (up to $t^m = 0.1$) the two schemes gave graphically indistinguishable results from each other. However for the coarser meshes as time evolved the results from the two schemes started to differ; in particular the meshes obtained using Scheme $\tilde{\text{A}}$ became rather distorted, while the meshes obtained using Scheme $\tilde{\text{B}}$ remained quite uniform, see Figure 10 in which results produced using Mesh 2 at $t^m = 0.26$ are displayed. From this figure we see that towards the top of the domain the elements in the mesh produced by Scheme $\tilde{\text{A}}$ (left hand plot) are elongated in the vertical direction and in order to have a better quality of mesh, remeshing is required, however the mesh arising from Scheme $\tilde{\text{B}}$ (right hand plot) has remained quite uniform and thus the need for remeshing is removed.

The effect that these different meshes have on the geometry of the surface can be seen in Figure 11; here the radial symmetry of the problem is used and plots of the radius of $\Gamma_h(t^m)$ obtained using Mesh 2 are displayed at $t^m = 0.06$ (left), $t^m = 0.16$ (centre) and $t^m = 0.26$ (right). In each plot the solution obtained using Scheme $\tilde{\text{A}}$ is given by a solid blue line and the solution for Scheme $\tilde{\text{B}}$ by a dashed red line. From this figure we see that for the early stages of motion (left hand plot) the two schemes give very similar results but as time evolves (centre plot) the profiles of the radii begin to differ. In particular the deformed mesh arising from Scheme $\tilde{\text{A}}$ (solid blue line) gives rise to a straight line section in the profile of the radius at the top of the domain. By $t^m = 0.26$ (right hand plot) the difference between the two radii increases, with the straight line section produced by Scheme $\tilde{\text{A}}$ becoming more pronounced. Figure 12 takes the same form as Figure 11 except that here the results were produced by Mesh 4 instead of Mesh 2. Here we see that at $t^m = 0.16$ the results from two schemes are now very similar however at the later time of $t^m = 0.26$ again a straight line section appears in the profile of the radius produced by Scheme $\tilde{\text{A}}$, so even for this fine mesh, remeshing is required to obtain a better quality mesh. We now use the results obtained from Mesh 4 to gauge how well the two schemes approximate the evolution of the interface when coarser meshes and larger time steps are used. We only consider $t^m \in [0, 0.16]$ since from Figure 12 we can see that in this time interval the results produced by Mesh 4 for the two schemes are almost identical. To this end we introduce a discrete approximation to the Hausdorff distance; we set

$$D_H(\Gamma_{h_1}, \Gamma_{h_2}) := \max\left(\max_{i\in\mathcal{I}_1}\min_{j\in\mathcal{I}_2}|\mathbf{X}_i - \mathbf{X}_j|, \max_{j\in\mathcal{I}_2}\min_{i\in\mathcal{I}_1}|\mathbf{X}_i - \mathbf{X}_j|\right)$$

to be a measure of how close two triangulated surfaces Γ_{h_1} and Γ_{h_2} are to each other. Here $\mathcal{I}_i$ is the set of vertices of Γ_{h_i}, $i = 1, 2$. Table 3 displays $D_H(\Gamma^m_{h_i}, \Gamma^m_{h4B})$ where Γ^m_{h4B} is the triangulated surface produced from Mesh 4 using Scheme $\tilde{\text{B}}$ and $\Gamma^m_{h_i}$ is the triangulated surface produced from Mesh i, for $i = 1, 2, 3$. From this table we see that for Meshes 1,2 and 3 the triangulated surfaces produced by Scheme $\tilde{\text{B}}$ have consistently lower values of $D_H(\Gamma^m_{h_i}, \Gamma^m_{h4B})$ than those produced by Scheme $\tilde{\text{A}}$. From the results in Section 3.4 it is fair to assume that since the more uniform meshes obtained from Scheme $\tilde{\text{B}}$ approximate the evolution of the surface better this scheme will also approximation the solution of the PDE on these surfaces better. Thus when coarser meshes and larger time steps are used we postulate that the results obtained from Scheme $\tilde{\text{B}}$ will be more accurate than the ones obtained from Scheme $\tilde{\text{A}}$.

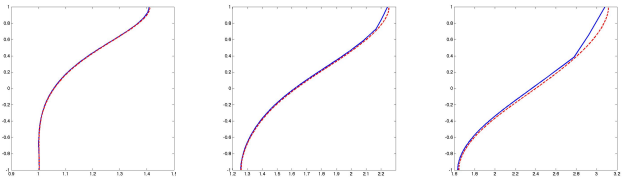

FIGURE 11. Results obtained from Mesh 2. Radius of $\Gamma_h(t^m)$ at $t^m = 0.06$ (left), $t^m = 0.16$ (centre) and $t^m = 0.26$ (right). Scheme $\tilde{\text{A}}$ (solid blue line) and Scheme $\tilde{\text{B}}$ (red dashed line).

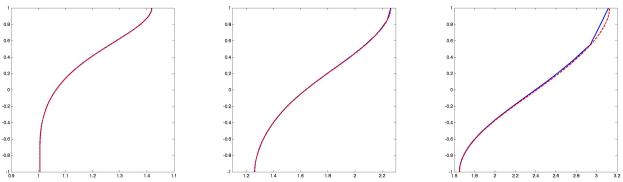

FIGURE 12. Results obtained from Mesh 4. Radius of $\Gamma_h(t^m)$ at $t^m = 0.06$ (left), $t^m = 0.16$ (centre) and $t^m = 0.26$ (right). Scheme $\tilde{\text{A}}$ (solid blue line) and Scheme $\tilde{\text{B}}$ (red dashed line).

Simulation 2

We set $\Omega = (-1, 1) \times (0, 4) \times (-1, 1)$, $\Gamma(0)$ to be the planar surface $x_2 = 0.1$, $U^0(x) \equiv 0$, $\boldsymbol{\nu}$ to be pointing in the increasing x_2 direction, $u_+ = 1$, $u_- = 0.5$, $\mathcal{D} = 1$, $\alpha = 5$ and $\beta = 5000$. Figure 13, contains results obtained

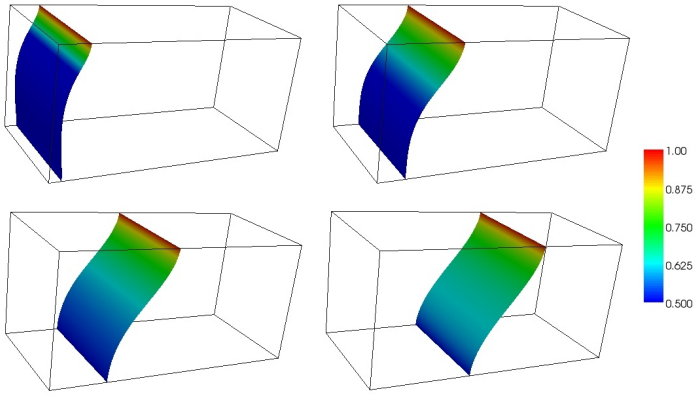

FIGURE 13. Simulation 2: U_h^m plotted on Γ_h^m at $t^m = 0.2, 0.50.8, 1.0$, (Scheme $\tilde{\text{B}}$).

using Scheme $\tilde{\text{B}}$, it displays the approximate solution $U_h(t^m)$ on $\Gamma_h(t^m)$ at times $t^m = 0.2, 0.5, 0.8, 1.0$. Since the interface is close to planar during the early stages of motion the the concentration term in (5.1b) dominates the motion. As the concentration of solute that diffuses in from the top is set to be twice that which diffuses from the bottom, the top of the interface moves faster than the bottom. This results in the curvature of Γ contributing more to its evolution. By the final subplot, $t^m = 1.0$, the concentration distribution and the shape of Γ_h do not change; in particular a travelling wave solution of the kind studied in [**?**] has been reached. We computed the problem on Meshes 1 - 4, see Table 2 Column C3, for both Scheme $\tilde{\text{A}}$ and Scheme $\tilde{\text{B}}$. For all of the meshes in the early stages of evolution (up to $t^m = 0.2$) the two schemes gave graphically indistinguishable results from each other. However for the coarser meshes as time evolved the results from the two schemes started to differ; in particular the meshes obtained using Scheme $\tilde{\text{A}}$ became rather distorted, while the meshes obtained using Scheme $\tilde{\text{B}}$ remained quite uniform, see Figure 14 in which results produced using Mesh 2 are displayed. In this figure the same section of the surface is shown for each scheme; we see that the mesh for Scheme $\tilde{\text{A}}$ (left hand plot) has become quite elongated in the vertical direction

	Scheme $\tilde{\text{A}}$ $t^m = 0.06$	Scheme $\tilde{\text{B}}$ $t^m = 0.06$	Scheme $\tilde{\text{A}}$ $t^m = 1.0$	Scheme $\tilde{\text{B}}$ $t^m = 1.0$	Scheme $\tilde{\text{A}}$ $t^m = 0.16$	Scheme $\tilde{\text{B}}$ $t^m = 0.16$
$D_H(\Gamma^m_{h_1}, \Gamma^m_{h_{4B}})$	1.81×10^{-2}	1.41×10^{-2}	8.63×10^{-2}	1.59×10^{-2}	4.45×10^{-2}	1.83×10^{-2}
$D_H(\Gamma^m_{h_2}, \Gamma^m_{h_{4B}})$	1.07×10^{-2}	7.15×10^{-3}	1.72×10^{-2}	8.20×10^{-3}	3.49×10^{-2}	9.46×10^{-3}
$D_H(\Gamma^m_{h_3}, \Gamma^m_{h_{4B}})$	1.03×10^{-2}	6.52×10^{-3}	4.15×10^{-3}	1.92×10^{-2}	1.08×10^{-2}	6.28×10^{-3}

TABLE 3. Simulation 1: Discrete Hausdorff distance between the triangulated surfaces $\Gamma^n_{h_{4B}}$ and $\Gamma^n_{h_i}$, $i = 1, 2, 3$.

while in different sections of the mesh (not shown) the elements become quite compacted in the vertical direction, however the mesh for Scheme $\tilde{\text{B}}$ (right hand plot) is quite uniform. As in Simulation 1, we studied the effect that the different meshes had on the geometry of the surface. Again we found that at the later times the distorted meshes produced by Scheme $\tilde{\text{A}}$ gave rise to straight line sections in the interfaces' profiles which did not arise in the profiles of the interfaces produced by Scheme $\tilde{\text{B}}$. For $t^m \in [0, 0.5]$ the results produced by Mesh 4 for the two schemes were almost identical. Taking the surfaces produced by one of these schemes, in particular the ones produced by Scheme $\tilde{\text{B}}$, to be a fair approximation of the true surfaces we approximate the Hausdorf distance between these surfaces and the surfaces produced by the coarser meshes. In Table 4 we present $D_H(\Gamma^m_{h_i}, \Gamma^m_{h_{4B}})$, where $\Gamma^m_{h_{4B}}$ is the surface produced by Scheme $\tilde{\text{B}}$ with Mesh 4 at t^m and $\Gamma^m_{h_i}$ are the surfaces produced by Schemes $\tilde{\text{A}}$ and $\tilde{\text{B}}$, with Mesh i, $i = 1, 2, 3$ at $t^m = 0.2, 0.4, 0.5$. From this table we see that, as in Simulation 1, for Meshes 1,2 and 3 the triangulated surfaces produced by Scheme $\tilde{\text{B}}$ have consistently lower values of $D_H(\Gamma^m_{h_i}, \Gamma^m_{h_{4B}})$ than those produced by Scheme $\tilde{\text{A}}$.

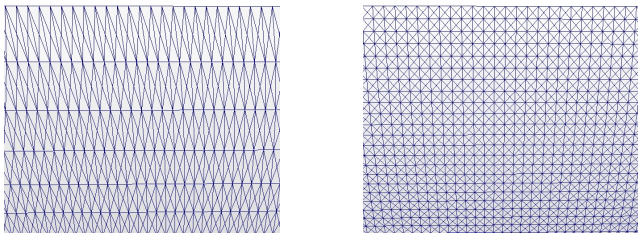

FIGURE 14. Comparison of the triangulated surfaces Γ^m_h at $t^m = 0.4$, Scheme $\tilde{\text{A}}$ (left) and Scheme B (right).

	Scheme $\tilde{\text{A}}$ $t^m = 0.2$	Scheme $\tilde{\text{B}}$ $t^m = 0.2$	Scheme $\tilde{\text{A}}$ $t^m = 0.4$	Scheme $\tilde{\text{B}}$ $t^m = 0.4$	Scheme $\tilde{\text{A}}$ $t^m = 0.5$	Scheme $\tilde{\text{B}}$ $t^m = 0.5$
$D_H(\Gamma^m_{h_1}, \Gamma^m_{h_{4B}})$	9.60×10^{-2}	9.74×10^{-2}	1.83×10^{-1}	1.67×10^{-1}	2.79×10^{-1}	2.53×10^{-1}
$D_H(\Gamma^m_{h_2}, \Gamma^m_{h_{4B}})$	4.93×10^{-2}	4.83×10^{-2}	7.55×10^{-2}	6.22×10^{-2}	1.28×10^{-1}	1.01×10^{-1}
$D_H(\Gamma^m_{h_3}, \Gamma^m_{h_{4B}})$	1.01×10^{-2}	5.61×10^{-3}	2.84×10^{-2}	1.52×10^{-2}	5.36×10^{-2}	2.56×10^{-2}

TABLE 4. Simulation 2: Discrete Hausdorff distance between the triangulated surfaces $\Gamma^n_{h_{4B}}$ and $\Gamma^n_{h_i}$, $i = 1, 2, 3$.

Simulation 3

We set $\Omega = (-1, 1) \times (0, 2) \times (-1, 1)$ and $\Gamma(0)$ to be the perturbed cylindrical surface $x_1^2 + x_2^2 = 0.9 + 0.04\cos(0.1 + 8\theta) + 0.05\cos(2\pi x_3)$, $-1 \leq x_3 \leq 1$, with ν being the inward pointing normal to $\Gamma_h(0)$ We took $U^0(x) = 0.2 + 0.05n(x)$, where $n(x) \in [0, 1)$ is random noise, and we set $u_+ = 1$, $u_- =$

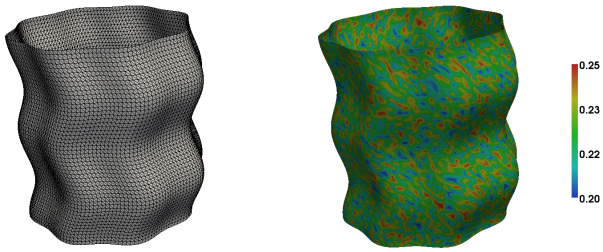

FIGURE 15. Simulation 3: Triangulated surface Γ_h^0 (left), U_h^0 plotted on Γ_h^0 (right).

0.5, $\mathcal{D} = 1$, $\alpha = 1$, $\beta = 1$, see Figure 15. We computed the problem on Meshes 1 - 4, see Table 2 Column C2. In this simulation for the coarser meshes Mesh 1 and Mesh 2, the complicated evolution of the geometry caused difficulties for both Scheme $\tilde{\text{A}}$ and Scheme $\tilde{\text{B}}$. This can be seen in Figure 16 in which the triangulated surfaces produced by the two schemes using Mesh 2 are displayed at time $t^m = 0.54$. The result from Scheme $\tilde{\text{A}}$ is shown on the left and the result from Scheme $\tilde{\text{B}}$ on the right. We see that the mesh produced by Scheme $\tilde{\text{A}}$ has stretched elements at the top of the domain, while the mesh produced by Scheme $\tilde{\text{B}}$ has stretched elements in the upper part of the domain on the left hand side. For both schemes the mesh distortion increases with time. These distortions were not produced by Scheme $\tilde{\text{B}}$ when the finer meshes, Mesh 3 and Mesh 4, were used, however they were still produced by Scheme $\tilde{\text{A}}$, although as expected the distortions reduced with the mesh size.

We conclude with Figure 17 in which we see how the distorted meshes produced by the two schemes effect the approximate solution $U_h(t^m)$. In particular we plot $U_h(t^m)$ on the triangulated surfaces displayed in Figure 16 together with plots of $U_h(t^m)$ on the undistorted meshes produced by Scheme $\tilde{\text{B}}$ using Mesh 4. The left hand plots show $U_h(t^m)$ on Γ_h^m produced by Scheme $\tilde{\text{A}}$ using Mesh 2, the center plots show $U_h(t^m)$ on Γ_h^m produced by Scheme $\tilde{\text{B}}$ using Mesh 2 and the right hand plots show $U_h(t^m)$ on Γ_h^m produced by Scheme $\tilde{\text{B}}$ using Mesh 4.

6. Concluding remarks

In this paper we have

- Formulated an ALE ESFEM on triangulated surfaces for the approximation of advection diffusion equations
- By numerical examples we have illustrated the effect of mesh quality on the accuracy of the solutions and values of using an ALE version of ESFEM.

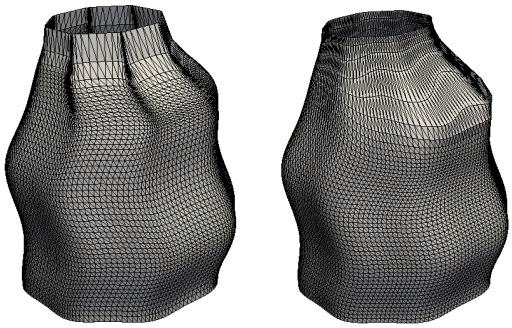

FIGURE 16. Simulation 3: Γ_h^m at $t^m = 0.54$, Scheme $\tilde{\text{A}}$ (left), Scheme $\tilde{\text{B}}$ (right).

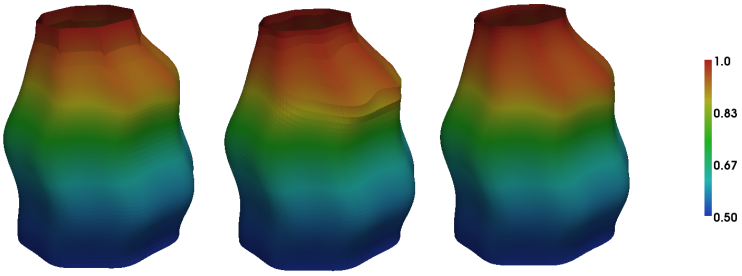

FIGURE 17. Simulation 3: U_h^m plotted on Γ_h^m at $t^m = 0.36, 0.54$, (vertically decreasing), Scheme $\tilde{\text{A}}$, Mesh 2 (left), Scheme $\tilde{\text{B}}$, Mesh 2 (centre), Scheme $\tilde{\text{B}}$, Mesh 4 (right).

- We have formulated an evolving surface finite element approach to the coupling of diffusion on a surface to the surface evolution on physical examples which show the ability of surface finite elements to simulate complex evolving geometries.

Charles M. Elliott
Mathematics Institute
University of Warwick
Coventry
CV4 7AL
UK
e-mail: c.m.elliott@warwick.ac.uk

Vanessa Styles
Department of Mathematics
University of Sussex,
Brighton
BN1 9QH
UK
e-mail: v.styles@sussex.ac.uk